

# **TUGAS AKHIR**

## **JUDUL**

### **KOSTUMISASI UBUNTU 9.10 UNTUK KEGIATAN PEMBELAJARAN BAHASA PEMROGRAMAN BERBASIS *OPEN SOURCE***

**(Studi Kasus : Jurusan Teknik Informatika UIN SUSKA)**

**Diajukan sebagai persyaratan mengikuti mata kuliah Tugas Akhir**

**pada Fakultas Sains dan Teknologi**

**Jurusan Teknik Informatika**

**UIN SUSKA RIAU**



**DIDI SURYADI**

**10551001449**

**FAKULTAS SAINS DAN TEKNOLOGI**

**JURUSAN TEKNIK INFORMATIKA**

**UIN SUSKA RIAU**

**2010**

# **KOSTUMISASI UBUNTU 9.10 UNTUK KEGIATAN PEMBELAJARAN BAHASA PEMROGRAMAN BERBASIS *OPEN SOURCE***

**(Studi Kasus : Jurusan Teknik Informatika UIN SUSKA)**

**DIDI SURYADI**

**10551001449**

Tanggal sidang : Juli 22<sup>th</sup> 2010

Periode Wisuda : November 30<sup>th</sup> 2010

Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Sultan Syarif Kasim Riau

## **ABSTRAK**

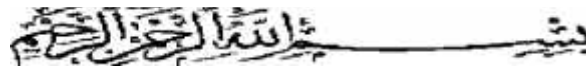
Kostumisasi merupakan salah satu teknik untuk mengembangkan sistem operasi *open source* menjadi sistem yang dapat menggabungkan berbagai aplikasi untuk tujuan tertentu. Kostumisasi memudahkan pengguna untuk menjalankan aplikasi dan sistem secara langsung.

Dalam tugas akhir ini, dilakukan penelitian pengembangan sistem operasi berisi paket bahasa pemrograman dengan cara kostumisasi, yang bersifat *open source*. Pengembangan ini untuk mengatasi permasalahan pada mahasiswa yang masih sulit menemukan paket aplikasi dan instalasi untuk bahasa pemrograman pada sistem operasi *open source*, sehingga terdapat satu sistem operasi yang memudahkan mahasiswa dalam kegiatan pembelajaran mata kuliah pemrograman. Pengembangan dilakukan dengan teknik *remastering*. Sistem operasi yang dihasilkan berisikan sejumlah aplikasi pemrograman dan aplikasi pendukung pemrograman lainnya. Penelitian ini dilakukan terhadap sistem operasi linux yaitu ubuntu 9.10.

Hasil akhir dari penelitian ini adalah sebuah sistem operasi ubuntu pemrograman yang dapat dipergunakan langsung aplikasinya, sehingga dapat dijadikan acuan dalam mempelajari aplikasi pemrograman *open source* di lingkungan Jurusan teknik Informatika UIN SUSKA.

**Kata Kunci :** Kostumisasi, Linux, *Open Source*, *Remastering*, Sistem Operasi, Ubuntu 9.10.

## KATA PENGANTAR



*Assalamu'alaikum Wr.Wb.*

Puji dan syukur saya sanjungkan kehadiran Allah SWT karena dengan rahmat dan hidayahnya penulis dapat menyelesaikan tugas akhir ini. Laporan tugas akhir yang diambil penulis berjudul ***“Kostumisasi Ubuntu 9.10 Untuk Kegiatan Pembelajaran Pemrograman Berbasis Open Source ”*** sebagai salah satu persyaratan kelulusan akademik pada Fakultas Sains dan Teknologi Universitas Islam Negeri Riau .

Keberhasilan penulisan laporan ini tidak lepas dari dukungan berbagai pihak, untuk itu pada kesempatan ini penulis ingin mengucapkan terima kasih kepada :

1. Allah SWT yang telah memberikan rahmat, karunia dan kemudahan dalam menyelesaikan tugas akhir ini.
2. Bapak Prof. Dr. H. M. Nazir, Selaku Rektor Universitas Islam Negeri Sultan Syarif Kasim Riau.
3. Ibu Dra. Hj. Yenita Morena, M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau
4. Bapak Jasril, S.Si, M.Sc, Ketua Jurusan Teknik Informatika.
5. Bapak Alwis Nazir, M. Kom selaku penguji satu dan Bapak Nazruddin Safaat, S.T selaku penguji dua.
6. Bapak Novriyanto, M.Sc selaku pembimbing satu dan Bapak Iwan Iskandar, S.T selaku pembimbing dua.
7. Ibu dan Ayahku tercinta, yang telah mendo'akan dan memberikan dukungan kepadaku agar dapat menyelesaikan laporan tugas akhir ini dengan baik.
8. Saudara-saudaraku tercinta, atas semangat dan dukungannya untuk segera menyelesaikan tugas akhir ini.

9. Teman-teman seperjuangan angkatan 2005 TIF lokal A, lokal B dan teman-teman yang lain yang tidak bisa penulis sebutkan satu persatu.
10. Teman-teman di yayasan Al-Ittihadiyah komplek Chevron Rumbai, yang memberikan bantuan, nasehat serta motivasi menuju sesuatu yang lebih baik dan bermamfaat.

Penulis menyadari bahwa dalam penulisan laporan ini masih banyak kesalahan dan kekurangan, oleh karena itu kritik dan saran yang sifatnya membangun sangat penulis harapkan untuk kesempurnaan laporan ini. Penulis dapat dihubungi di [suryadididi86@yahoo.com](mailto:suryadididi86@yahoo.com). Akhirnya penulis berharap semoga laporan ini dapat memberikan informasi yang bermanfaat bagi siapa saja yang membacanya.

Wassalamu'alaikum Wr.Wb

Pekanbaru, 22 Juli 2010

Penulis

DIDI SURYADI

## DAFTAR ISI

	Halaman
LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN .....	iii
LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL.....	iv
LEMBAR PERNYATAAN.....	v
LEMBAR PERSEMBAHAN .....	vi
ABSTRAK.....	vii
<i>ABSTRACT</i> .....	viii
KATA PENGANTAR .....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR .....	xiv
DAFTAR TABEL.....	xv
BAB I PENDAHULUAN .....	I-1
1.1 Latar Belakang .....	I-1
1.2 Rumusan Masalah .....	I-2
1.3 Batasan Masalah.....	I-2
1.4 Tujuan Penelitian .....	I-3
1.5 Manfaat Penelitian .....	I-3
1.6 Sistematika Penulisan .....	I-3
BAB II LANDASAN TEORI.....	II-1
2.1 Sistem Operasi .....	II-1
2.2 Tujuan Sistem Operasi .....	II-2
2.3 Sistem Operasi Linux.....	II-2
2.3.1 <i>Kernel Linux</i> .....	II-2
2.3.2 <i>System Library</i> .....	II-3
2.3.3 <i>System utility</i> .....	II-4
2.4 Struktur Direktori Linux .....	II-6
2.5 Metode Pengembangan Linux.....	II-7

2.6 Model Proses Sekuential Linier .....	II-10
2.7 Sistem Operasi Ubuntu .....	II-11
2.8 Kelebihan dan Kerugian Ubuntu.....	II-12
 BAB III METODOLOGI PENELITIAN.....	 III-1
3.1 Tahapan Penelitian .....	III-1
3.2 Jadwal Pengerjaan .....	III-2
3.3 Pengumpulan Data .....	III-2
3.4 Analisa Sistem.....	III-3
3.5 Perancangan Sistem Operasi <i>Remastering</i> .....	III-3
3.6 Implementasi .....	III-4
3.7 Pengujian Sistem.....	III-4
 BAB IV ANALISA DAN PERANCANGAN .....	 IV-1
4.1 Analisa Sistem Lama.....	IV-1
4.2 Analisa Sistem Baru .....	IV-3
4.2.1. Analisa Data .....	IV-4
4.2.2. Analisa <i>Metode</i> .....	IV-5
4.2.3 Analisa <i>Hardware</i> .....	IV-5
4.3 Perancangan <i>Remastering</i> Ubuntu 9.10 .....	IV-7
 BAB V IMPLEMENTASI DAN PENGUJIAN .....	 V-1
5.1 Implementasi .....	V-1
5.2 Batasan Implementasi .....	V-1
5.3 Lingkungan Implementasi.....	V-1
5.4 Hasil Implementasi.....	V-1
5.5 Pengujian.....	V-7
5.6 Rencana Pengujian .....	V-7
5.7 Hasil Pengujian .....	V-7
5.8 Kesimpulan Pengujian .....	V-8

BAB VI PENUTUP.....	VI-1
6.1 Kesimpulan .....	VI-1
6.2 Saran.....	VI-1
DAFTAR PUSTAKA	
LAMPIRAN	
DAFTAR RIWAYAT HIDUP	

## DAFTAR TABEL

## DAFTAR TABEL

Tabel	Halaman
2.1 Isi /usr/src/linux .....	II-4
2.2 Direktori Linux .....	II-7
2.3 Daftar Rilis Ubuntu .....	II-12
3.1 Jadwal pengerjaan Tugas Akhir .....	III-2
3.2 Perbandingan Desktop Environtment .....	III-3
4.1 Kebutuhan Aplikasi Pemrograman .....	IV-3
4.2 Perbandingan Desktop GNOME dan KDE .....	IV-3
4.3 Informasi <i>hardware</i> .....	IV-6
5.1 Perbandingan aplikasi <i>Ubuntu 9.10 dengan remastering</i> .....	V-2
5.2 Pengujian Instalasi Sistem Operasi Remastering .....	V-7
5.3 Pengujian Aplikasi Pemrograman .....	IV-7



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Kostumisasi merupakan salah satu teknik pengembangan sistem operasi *open source*, yang bertujuan untuk membuat sistem sesuai dengan kebutuhan. Seperti pengembangan sistem operasi dalam bidang multimedia, *networking*, pemrograman, pendidikan dan lainnya. Pengembangan dengan cara kostumisasi memberikan kemudahan dalam pemakaian sistem tersebut, karena dibuat dengan tujuan tertentu serta sesuai dengan kebutuhan. Kostumisasi sistem operasi banyak dilakukan pada linux ubuntu, seperti pada pengembangan *networking*, multimedia, dan bidang pendidikan.

Sistem operasi linux ubuntu, banyak digunakan di berbagai bidang, seperti bidang pemerintahan, perusahaan dan pendidikan. Permasalahan yang biasanya dihadapi oleh pengguna sistem operasi *open source* linux adalah sulitnya menemukan sistem operasi dan aplikasi yang sesuai kebutuhan, teknik instalasi yang bergantung pada beberapa *file* lainnya (*redundance file*), sehingga membuat sistem operasi linux lambat berkembang.

UIN SUSKA Riau khususnya Fakultas Sains dan Teknologi jurusan Teknik Informatika, pada kegiatan belajar mengajar seperti *networking*, multimedia, pemrograman juga ada yang menggunakan sistem operasi *open source* linux ubuntu dan tidak terlepas dari permasalahan diatas. Terutama pada bidang pemrograman, karena pada bidang ini, belum ditemukan sistem operasi yang menyediakan langsung aplikasi pemrograman berbasis *open source*.

Salah satu solusi untuk menyelesaikan masalah pada sistem operasi linux ini adalah dengan cara melakukan kostumisasi sistem operasi linux, berdasarkan kebutuhan paket aplikasi yang diinginkan, terutama dalam bidang bahasa pemrograman. Selain itu dapat juga dilakukan optimasi kernel untuk meningkatkan kemampuan sistem untuk dapat mendeteksi *hardware* dengan baik.

Beberapa sistem linux ubuntu yang sudah dilakukan kostumisaasi misalnya :

- a. Linux ubuntu studio, kostumisasi dalam bidang multimedia yang menyediakan berbagai aplikasi untuk kebutuhan multimedia.
- b. Linux ubuntu *server*, kostumisasi dalam bidang jaringan, yang menyediakan aplikasi untuk kebutuhan jaringan.

Sedangkan pada sistem linux pemrograman, belum dilakukan kostumisasi, oleh sebab itu perlu adanya sistem yang memaketkan sistem operasi linux dengan bahasa pemrograman. Oleh sebab itu perlu dilakukan kostumisasi sistem operasi linux dalam bidang pemrograman *open source* yang sudah dilakukan optimasi kernel.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang diatas, maka didapat rumusan masalah yaitu, Bagaimana melakukan Kostumisasi Linux Ubuntu dengan menyediakan aplikasi pemrograman berbasis *Open Source* dan aplikasi pendukung lainnya.

## **1.3 Batasan Masalah**

Adapun batasan masalah dari penelitian ini adalah:

1. Kostumisasi sistem linux ini menggunakan kernel Ubuntu 9.10 versi 2.6.31.14 bersifat *generic*, dapat digunakan hampir diseluruh *hardware*.

2. Kostumisasi dari segi desktop menggunakan *GNOME* sebagai *Desktop Environment*.
3. Optimasi kernel menggunakan *GNU C Compilers*. Optimasi kernel dilakukan pada beberapa *hardware* seperti : *CPU, APM Power Manager, Bus Option, Device Driver, ACPI*.

#### **1.4 Tujuan Penelitian**

Tujuan penelitian ini adalah :

1. Membangun aplikasi dan bahasa pemrograman *open source* pada sistem operasi linux ubuntu 9.10.
2. Melakukan optimasi pada kernel, dengan melakukan *compile kernel* untuk kelancaran dalam pengenalan *device* komputer.

#### **1.5 Manfaat Penelitian**

Manfaat penelitian yang dilakukan adalah :

1. Sistem operasi yang dihasilkan dapat digunakan sebagai media pembelajaran bahasa pemrograman *Open Source* di jurusan Teknik Informatika UIN Suska Riau.
2. Alternatif sistem operasi yang dapat digunakan sebagai *software* pendukung praktek mahasiswa di laboratorium pemrograman.

#### **1.6 Sistematika Penulisan**

Dalam penyusunan laporan kerja praktek ini, penulis membaginya dalam beberapa bab agar lebih sistematis dan terarah. Bab-bab yang disajikan adalah sebagai berikut:

1. **BAB I PENDAHULUAN**; bagian ini berisi tentang deskripsi umum tugas akhir yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematika penulisan.
2. **BAB II LANDASAN TEORI**; bagian ini menjelaskan tentang teori-teori umum, teori-teori khusus yang berhubungan dengan tugas akhir ini.
3. **BAB III METODOLOGI PENELITIAN**; bagian ini menjelaskan tentang urutan kerja yang dilakukan dalam penelitian, seperti pengumpulan data data, analisa, perancangan, tempat serta waktu penelitian.
4. **BAB IV ANALISA DAN PERANCANGAN**; bagian ini berisi tentang analisis dan perancangan sistem operasi. Pada tahap perancangan akan dibahas tentang teknik, proses perancangan dan juga isi dari sistem operasi yang akan dibangun.
5. **BAB V IMPLEMENTASI DAN PENGUJIAN**; pada bagian implementasi dan pengujian berisi pembahasan mengenai implementasi sistem operasi disertai dengan *preview* tampilan *interface* serta pengujian.
6. **BAB VI KESIMPULAN DAN SARAN**; bagian ini berisi kesimpulan hasil dari semua tahap yang telah dilalui selama penelitian beserta saran-saran yang berkaitan dengan penelitian ini.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Kostumiasi Sistem Operasi linux**

Kostumisasi merupakan salah satu teknik penyesuaian sistem operasi dengan aplikasi untuk disatukan menjadi sistem yang dirancang sesuai dengan kebutuhan. Tujuan kostumisasi adalah : Merancang sistem operasi *open source* sesuai dengan kebutuhan, menyesuaikan *hardware* yang ada sehingga dapat digunakan pada sistem.

#### **2.2 Tujuan Sistem Operasi**

Tujuan mempelajari sistem operasi adalah (Hariyanto, 1999) ;

1. Agar dapat merancang sendiri atau memodifikasi sistem operasi yang telah ada sesuai kebutuhan.
2. Agar dapat menilai sistem operasi dan memilih alternatif sistem operasi sesuai tujuan pembangunan sistem.
3. Pemakai harus berinteraksi dengan sistem komputer untuk menyelesaikan tugasnya karena sistem operasi merupakan antarmuka utama dalam berhubungan dengan sistem operasi.

Sedangkan menurut Stalling tujuan sistem operasi adalah (Stalling, 2001) :

1. Kenyamanan, membuat *user* lebih nyaman dalam pemakaian sistem.
2. Efisien, penggunaan sumber daya secara efisien.
3. Mampu berevolusi seperti memudahkan pengembangan, pengujian, dan pengajuan fungsi baru.

## 2.3 Sistem Operasi Linux

Sistem Operasi GNU Linux terdiri dari tiga bagian kode penting yaitu:

1. *Kernel Linux*.
2. *System Library*.
3. *System Utility*.

### 2.3.1. *Kernel Linux*

Kernel merupakan inti dari suatu sistem operasi yang bertugas sebagai pengendali dan mengontrol kinerja dari semua yang ada pada sistem, mulai dari peralatan, penggunaan memori, sampai pada aplikasi berjalan. Kernel juga mengatur sumber daya yang digunakan mengatur peletakan *file*, mengenali *driver*, dan hal lainnya.

Kernel memiliki *Module*, bagian *code* kernel yang secara tidak langsung dimasukkan ke dalam kernel. *Module* dapat dimasukkan atau dihilangkan kedalam kernel yang sedang berjalan kapanpun diperlukan. Selain itu kernel juga memiliki layanan yang disebut *System Call*, berfungsi untuk mengimplementasikan layanan yang dimiliki sistem.

Adapun tugas utamanya adalah :

1. Manajemen *Input* dan *Output*
2. Manajemen Proses
3. Manajemen Perangkat Keras
4. Manajemen *File*
5. Manajemen Memori

Kernel linux terletak di `/usr/src/linux` :

**Tabel 2.1 isi dari /usr/src/linux**

<b>Nama file</b>	<b>Keterangan</b>
Documentation	Berisi informasi tentang kernel Linux dan subsistemnya. Beberapa informasi diperuntukkan bagi <i>end user</i> (pengguna). Selain itu, terdapat juga beberapa <i>file</i> yang khusus diperuntukkan bagi developer Linux.
<i>Arch</i>	Tipe dari arsitektur <i>processor</i> komputer yang didukung oleh kernel Linux. isi dari sub direktori tersebut antara lain: alpha cris ia64 mips parisc s390 sh sparc64 arm i386 m68k mips64 ppc s390x sparc.
<i>Drivers</i>	Berisi kode-kode program <i>driver</i> perangkat keras seperti <i>driver USB drive</i> , kartu jaringan, <i>video card</i> , . fs: Berisi kode program yang digunakan untuk mendukung <i>file system</i> yang ada (misal: ntfs, ext2, ext3, msdos, minix, )
<i>Include</i>	Berisi <i>header file</i> (*.h) yang akan digunakan oleh <i>source</i> kernel
<i>Init</i>	Kode inisialisasi kernel.
<i>Ipc</i>	Kode kode yang diimplementasikan <i>System V InterProcess Communication APIs</i> ( <i>semaphores, message queues, shared memory</i> ).
<i>Kernel</i>	Berisi kode-kode dasar dari kernel Linux. Berisi kode program untuk proses penjadwalan ( <i>scheduling</i> ), pengalokasian <i>resource</i> , modul-modul kernel.
<i>Lib</i>	Standar pustaka C (misal: ctype, sprintf)
<i>Mm</i>	Berisi kode sumber untuk sistem manajemen memori Linux.
<i>Net</i>	Berisi kode-kode program untuk diimplementasikan pada protokol jaringan
<i>Scripts</i>	Berisi <i>script-script</i> untuk konfigurasi kernel (misal: xconfig)

### 2.3.2. System Library

*Library System* menyediakan banyak tipe fungsi. Pada level yang paling rendah, diijinkan aplikasi melakukan permintaan pada *service* sistem kernel. Membuat sebuah panggilan sistem (*system calls*) melibatkan transfer kontrol dari *mode user* yang tidak penting ke *mode kernel* yang penting, rincian dari transfer ini berbeda pada masing-masing arsitektur.

*Library* bertugas untuk mengumpulkan argumen *system-call* dan jika perlu, mengatur argumen tersebut dalam bentuk khusus yang diperlukan untuk melakukan *system call*. *Library* juga dapat menyediakan versi lebih kompleks dari *system call* dasar.

Contohnya, *fungsi buffered file-handling* dari bahasa C semuanya diimplementasi dalam *library system*, yang berakibat kontrol yang lebih baik terhadap *file input output* daripada yang disediakan oleh *system call* kernel dasar. *Library* juga menyediakan rutin yang tidak ada hubungan dengan *system call*, seperti algoritma penyusunan (*sorting*), fungsi matematika, dan rutin manipulasi string (*string manipulation*).

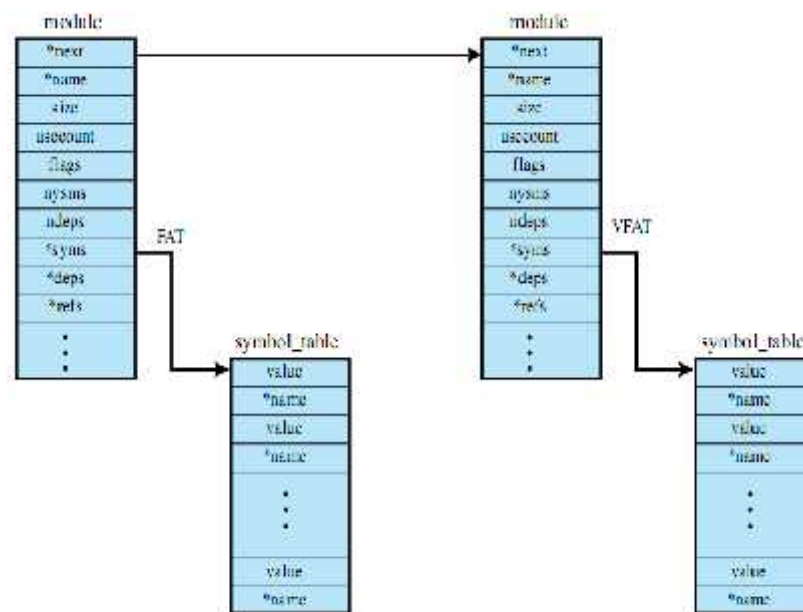
### **2.3.3. *System Utility***

Sistem linux mengandung banyak program-program *user-mode*, *utility sistem* dan *utility user*. *Utility sistem* termasuk semua program yang diperlukan untuk menginisialisasi sistem, seperti program untuk konfigurasi alat jaringan (*network device*) atau untuk *load* modul kernel. Program *server* yang berjalan secara *kontinu* juga termasuk sebagai utilitas *system*, program semacam ini mengatur permintaan *user login*, koneksi jaringan yang masuk, dan antrian printer. Tidak semua utilitas standar melakukan fungsi administrasi sistem yang penting.

Lingkungan pengguna UNIX mengandung utilitas standar dalam jumlah besar untuk melakukan pekerjaan sehari-hari, seperti membuat daftar direktori, memindahkan dan menghapus *file*, atau menunjukkan isi dari sebuah *file*. Utilitas yang lebih kompleks dapat melakukan fungsi *textprocessing*, seperti menyusun data tekstual atau melakukan *pattern-searches* pada input teks. Jika digabung, utilitas-utilitas tersebut membentuk *toolset* standar yang diharapkan oleh *user* pada sistem UNIX mana saja; walau pun tidak melakukan fungsi sistem operasi apa pun, *utility* tetap merupakan bagian penting dari sistem Linux dasar.



Menurut Stalling (Stalling, 2001) struktur sistem linux bekerja berdasarkan *modular structure* pada kernelnya. Modul-modul tersebut disertakan dalam *source* kernel linux, sehingga dapat ditambah dan dikurangi. Modul-modul pada kernel tersebut dapat di *load* secara langsung bersama kernel atau bisa juga bersifat opsional yaitu modul dijalankan secara manual oleh pengguna. Adapun contoh struktur modul kernel pada linux menurut stalling adalah sebagai berikut:

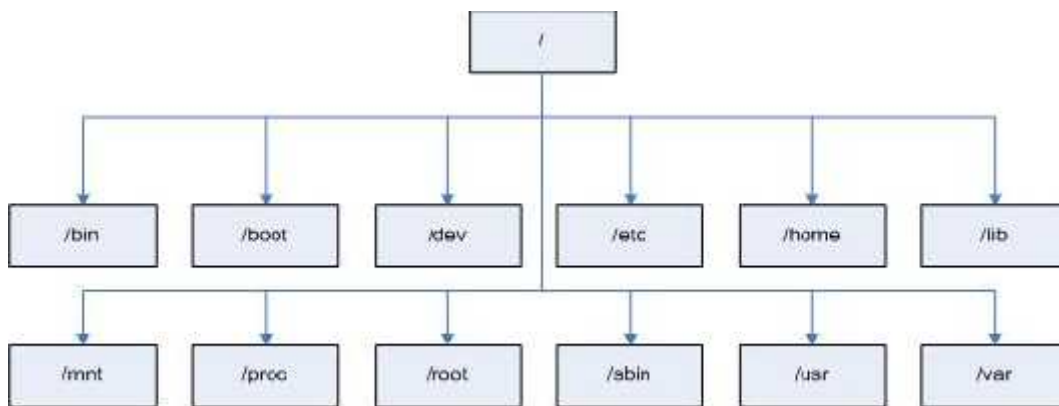


Gambar 2.1 Struktur Model Kernel Linux

Gambar diatas menunjukkan beberapa modul yang diproses diatas kernel, satu tabel pada gambar diatas merupakan satu proses yang didalamnya terdapat beberapa proses yang diberi tanda panah.

## 2.3 Struktur Direktori Linux

Direktori GNU Linux memiliki beberapa direktori yang merupakan standar direktori pada banyak distro GNU Linux. Berikut ini direktori bawaan dari linux pada saat instalasi ( Andreas, 2003) :



Gambar 2.2 Struktur Direktori Linux

Tabel 2.2 Direktori Linux

Direktori	Isi
<i>/bin</i>	berisi <i>file-file binary</i> standar yang dapat digunakan oleh seluruh <i>user</i> baik user biasa maupun super user.
<i>/boot</i>	berisi <i>file-file</i> yang digunakan untuk <i>booting</i> Linux termasuk kernel image.
<i>/dev</i>	berisi file system khusus yang merupakan refleksi <i>device hardware</i> yang dikenali dan digunakan <i>system</i> .
<i>/etc</i>	berisi <i>file-file</i> konfigurasi sistem, biasanya hanya boleh diubah oleh super user
<i>/home</i>	berisi direktori-direktori yang merupakan direktori home untuk <i>user</i> biasa dan aplikasi tertentu.
<i>/lib</i>	berisi <i>file-file library</i> yang digunakan untuk mendukung kerja kernel Linux.
<i>/mnt</i>	direktori khusus yang disediakan untuk mounting (mengaitkan) <i>device</i> disk storage ke sistem dalam bentuk direktori.
<i>/proc</i>	berisi <i>file system</i> khusus yang menunjukkan data-data kernel setiap saat.
<i>/root</i>	direktori home untuk <i>user root</i> (user khusus dengan priviledges hampir tak terbatas).
<i>/sbin</i>	sama seperti direktori bin, tetapi hanya super user yang sebaiknya menggunakan <i>binary-binary</i> tersebut mengingat fungsifungsi binary yang terdapat di direktori ini untuk maintenance system.
<i>/tmp</i>	berisi <i>file-file</i> sementara yang dibutuhkan sebuah aplikasi yang sedang berjalan.
<i>/usr</i>	berisi <i>library, binary</i> , dokumentasi dan file lainnya hasil instalasi <i>user</i> .
<i>/var</i>	berisi <i>file-file log</i> , mailbox dan data-data aplikasi.

## 2.4 Metode Pengembangan Linux

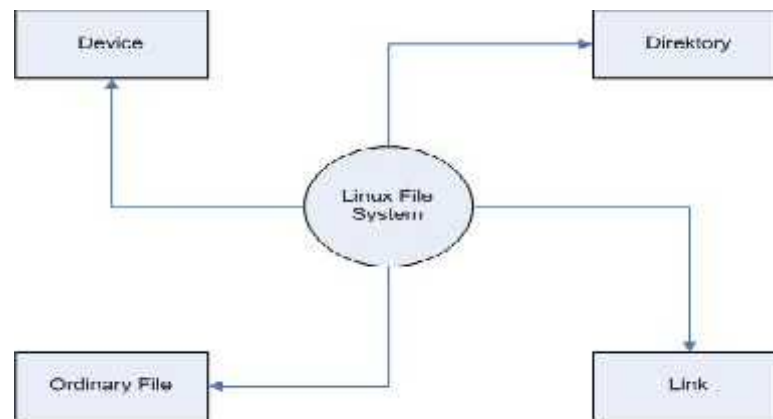
Dalam mengembangkan sistem operasi linux ada 2 metode (Sofyan, 2006) antara lain :

1. *Linux From Scratch (LFS).*
2. *Remastering.*

*Linux from Scratch (LFS)*, yaitu teknik perancangan atau pemaketan aplikasi kedalam distribusi GNU Linux yang dimulai dari nol. Metode ini dimulai dari pembuatan partisi, *patches*, pengambilan *kernel*, pembuatan *modul*, *driver*, *aplikasi*, *development tools*, *Compiler tool*, setelah itu tiap-tiap *tools* tersebut dikonfigurasi sesuai dengan tujuan pembuatan.



Gambar 2.3 Alur Proses *Linux From Scratch*



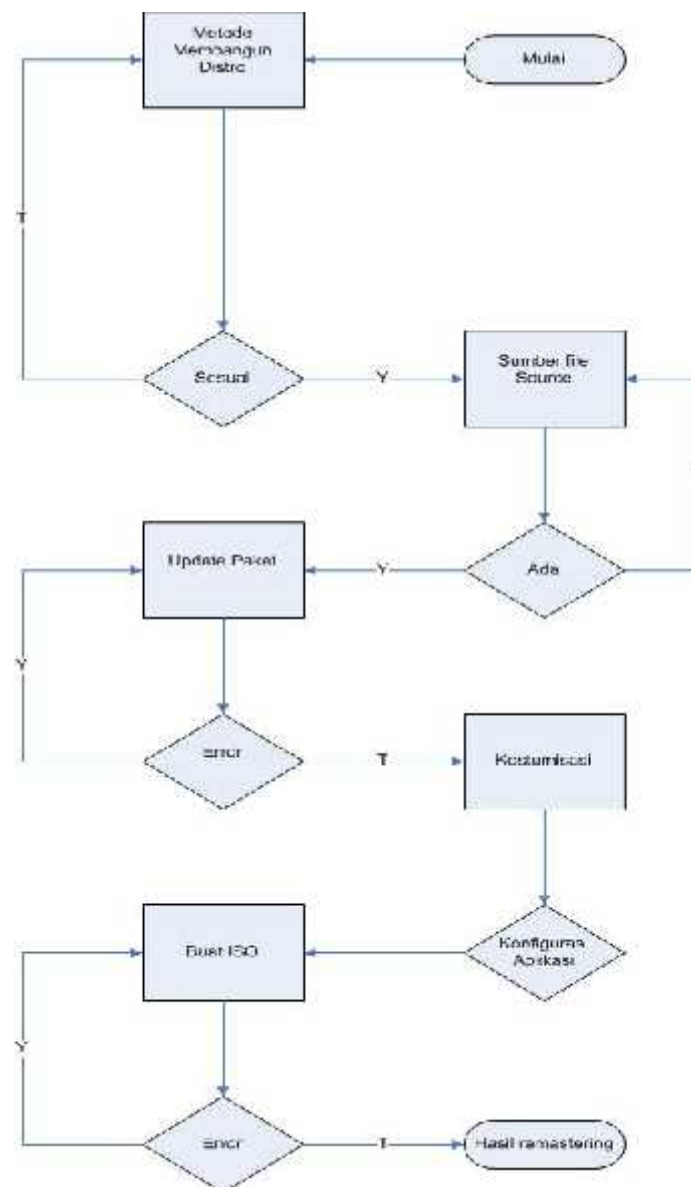
#### 2.4 Komponen *File Linux from Scratch*

Gambar diatas merupakan struktur *file system* dalam *LFS*, berbeda dengan struktur linux dalam proses *remastering* yang sama dengan struktur dasar dari linux yang di kembangkan.

Adapun fungsi dari komponen LFS tersebut adalah :

1. *Ordinary file* berfungsi sebagai *file* utama linux.
2. *Device* berfungsi untuk menangani *hardware* yang dikendalikan oleh linux.
3. *Direktory* berfungsi tempat dasar pengelompokan *file-file* linux.
4. *Link* berfungsi sebagai representasi *file* asli atau *file* dasar.

*Remastering* yaitu teknik perancangan distribusi GNU Linux dengan cara memecah isi (*unpack iso*) sistem operasi GNU Linux kedalam bentuk struktur direktori, tiap-tiap struktur direktori dikonfigurasi berdasarkan fungsinya, baik berupa *remove*, *add*, *update*, *upgrade*, *modul and driver*, dan hal lainnya, yang dilangsung dijalankan ketika proses terjadi, kemudian dibentuk kembali menjadi file iso.



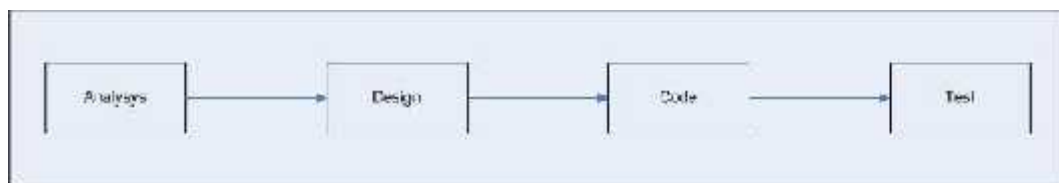
Gambar 2.5 Flowchart Remastering Linux

## 2.5 Model Proses Sekuensial linier

Pada penelitian ini model proses yang digunakan adalah model *sekuensial linier* yakni pengembangan perangkat lunak sistematis dan sekuensial. Penulis menggunakan model proses ini dikarenakan pada saat pengembangan variant linux ini tidak berinteraksi pada *user*.

Adapun aktifitas yang dilakukan pada model proses sekuensial linier (Pressman, 2002) antara lain :

1. Analisa (*Analysis*) kebutuhan perangkat lunak, pada tahap ini dilakukan analisa kebutuhan terhadap perangkat lunak yang akan dikembangkan.
2. Desain (*Design*), pada tahap ini dilakukan proses pemodelan dari pengembangan perangkat lunak ini yang akan dilakukan.
3. Generasi Code (*Code Generation*), pada tahap ini dilakukan proses untuk meng-*generate* code dan melakukan proses kompilasi.
4. Pengujian (*Testing*), pada tahap ini dilakukan proses testing terhadap perangkat lunak yang dihasilkan.



Gambar 2.6 Model Proses sekuensial Linier

## 2.6 Sistem Operasi Ubuntu

Ubuntu merupakan turunan dari debian. Tujuan dari distribusi Linux Ubuntu adalah membawa semangat yang terkandung di dalam Ubuntu ke dalam dunia perangkat lunak. Ubuntu adalah sistem operasi lengkap berbasis Linux, tersedia secara bebas dan mempunyai dukungan baik yang berasal dari komunitas maupun tenaga ahli profesional. Ubuntu berkembang sangat cepat karena dukungan yang baik dan menyediakan berbagai kebutuhan aplikasi hampir diseluruh bidang computer. Berikut ini daftar rilis linux ubuntu :

**Tabel 2.3 Daftar rilis distribusi GNU Linux ubuntu**

Versi	Tanggal Rilis	Kode	Sub versi	Waktu Dukungan
4.10	20 Oktober 2004	Warty Warthog	Sounder	30 April 2006
5.04	8 April 2005	Hoary Hedgehog	Array	31 Oktober 2006
5.10	13 Oktober 2005	Breezy Badger	Colony	13 April 2007
6.06	1 Juni 2006	Dapper Drake	Flight Juni 2009 desktop Juni 2011 server	Juni 2009 (desktop) Juni2011 (server)
Versi	Tanggal Rilis	Kode	Sub versi	Waktu Dukungan
6.10	26 Oktober 2006	Edgy Eft	Knot	April 2008
7.04	19 April 2007	Feisty Fawn	Herd	Oktober 2008
7.10	18 Oktober 2007	Gutsy Gibbon	Tribe	April 2009
8.04	21 April 2008	Hardy Heron	Siege	April 2011 (desktop) April 2013 (server)
8.10	Oktober 2008	Intrepid Ibex	8.10	Oktober 2010
9.04	April 2009	Jaunty Jackalope	9.04	April 2011
9.10	Oktober 2009	Karmic Koala	9.10	Oktober 2011

## 2.7 Kelebihan dan kekurangan Ubuntu

Berikut ini merupakan kelebihan di dalam sistem operasi ubuntu :

1. Ubuntu bersifat *open source*, artinya produk – produk ubuntu boleh untuk dimiliki secara bebas, yang dapat di tambah atau di kurangi, dimiliki, dan digandakan.
2. Bantuan komunitas, Ubuntu merupakan salah satu sistem operasi *open source* yang memiliki komunitas terbanyak di dunia, dengan banyaknya bantuan komunitas ini, maka segala permasalahan dapat segera terselesaikan.
3. Mudah digunakan, Ubuntu sangat mudah digunakan dikarenakan sudah memiliki *grafical user Interface* (GUI) yang baik.
4. Ketersediaan Repository, Repository merupakan kumpulan aplikasi yang hanya digunakan untuk distro khusus saja. Aplikasi ini dapat dikumpulkan di dalam media penyimpanan.
5. Dukungan Perusahaan, Ubuntu didukung oleh salah satu perusahaan terbesar di dunia yaitu Canonical Ltd.

Sedangkan menurut beberapa dari pengguna Ubuntu, ada beberapa kekurangan dari sistem operasi ini diantaranya :

1. Komunitas terbanyak hanya berada pada negara-negara maju, sedangkan pada negara berkembang kebanyakan menerima hasil dan tidak banyak ikut berpartisipasi terhadap perkembangan Ubuntu.
2. Tidak semua *device* komputer di dukung langsung oleh ubuntu. Sehingga membutuhkan keahlian khusus untuk mendeteksinya *device* yang ada.
3. Dokumentasi pada beberapa aplikasi yang paling banyak digunakan oleh negara berkembang seperti aplikasi perkantoran, penggunaan grafik, pengeditan gambar dinilai masih kurang.



## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Tahapan Penelitian

Metodologi penelitian digunakan sebagai pedoman dalam pelaksanaan penelitian agar hasil yang dicapai tidak menyimpang dari tujuan. Berikut tahapan penelitian :



Gambar 3.1 Tahapan Penelitian

### 3.2 Jadwal Pengerjaan

Jadwal pengerjaan Tugas Akhir ini dijelaskan pada tabel 3.1 di bawah ini.

**Tabel 3.1 Jadwal Pengerjaan Tugas Akhir**

No	Kegiatan	Waktu pelaksanaan					Agustus
		Februari	Maret	April	Juni	Juli	
1	Studi kepustakaan						
2	Analisa dan pengumpulan data						
3	Perancangan dan Pembuatan						
4	Implentasi dan Pengujian						
5	Laporan dan rekomendasi						

### 3.3 Pengumpulan Data

Dalam melakukan penelitian tugas akhir ini, adapun tahap pengumpulan data yang dilakukan adalah :

1. Studi Pustaka

Studi Pustaka yang dilakukan untuk mendukung penelitian yang akan dikerjakan bersumber dari buku-buku, jurnal, internet dan penelitian yang sejenis.

2. Wawancara

Wawancara dilakukan terhadap dosen pada bidang bahasa pemrograman di jurusan Teknik Informatika UIN Suska Riau.

### 3.4 Analisa Sistem

Dari hasil pengumpulan data yang diperoleh, selanjutnya dilakukan analisa. Analisa dilakukan dengan 2 tahap, tahap pertama analisa pada sistem lama dan tahap kedua pada analisa sistem baru.

Berikut ini analisa yang akan dilakukan secara bertahap :

1. Analisa sistem lama

Menganalisa sistem yang digunakan seperti sistem operasi, aplikasi, bahasa pemrograman yang digunakan, serta mata kuliah yang berhubungan dengan bahasa pemrograman.

2. Analisa sistem baru

Menganalisa sistem yang akan dirancang, seperti metode perancangan yang digunakan, aplikasi dan bahasa pemrograman yang dibutuhkan dalam mata kuliah pemrograman, jenis *device* yang ada untuk sistem yang baru.

### 3.5 Perancangan Sistem Operasi *Remastering*

Pada tahap awal perancangan sistem operasi ini, penulis melakukan beberapa tahap kegiatan, yaitu :

- a. Mendapatkan *file source* (Aplikasi yang dibutuhkan).
- b. Konfigurasi ulang seperti :
  1. Modifikasi pada desktop (*Theme, wallpaper, icon, panel*).
  2. Penambahan aplikasi Pemrograman.
  3. Pengurangan aplikasi yang tidak diperlukan.
- c. Melakukan optimasi ubuntu dengan Kompilasi Kernel.

### 3.6 Implementasi

Implementasi merupakan tahap dimana sistem operasi siap untuk di operasikan sesuai dari hasil analisis dan perancangan yang telah dilakukan, sehingga akan diketahui apakah sistem atau aplikasi yang dirancang benar-benar dapat menghasilkan tujuan yang ingin dicapai.

Sistem Operasi	: Linux Ubuntu 9.10 <i>Programming</i>
<i>Memory</i>	: 512 MB
<i>Harddisk</i>	: 160 GB
<i>Kernel</i>	: 2.6.31-14- <i>generic</i>

### 3.7 Pengujian Sistem Operasi *Remastering*

Tahap pengujian dilakukan dengan metode *black box*, untuk menjamin sistem operasi yang dibuat sesuai dengan hasil analisis dan perancangan serta menghasilkan satu kesimpulan apakah sistem tersebut sesuai dengan yang diharapkan.

## **BAB IV**

### **ANALISA DAN PERANCANGAN**

Analisa merupakan langkah pemahaman persoalan sebelum mengambil tindakan atau keputusan penyelesaian hasil utama. Analisa adalah proses untuk penyelesaian permasalahan terhadap kumpulan data-data yang bernilai informasi dan berguna untuk membentuk sistem.

#### **4.1 Analisa Sistem Lama**

Penggunaan bahasa pemrograman dijurusan Teknik Informatika UIN Suska Riau, disesuaikan dengan mata kuliah yang tersedia. Pada sistem lama ini analisa meliputi sistem operasi yang digunakan, bahasa pemrograman dan aplikasi yang digunakan untuk mempelajari bahasa pemrograman.

Sistem operasi yang digunakan adalah sistem operasi *windows xp*, sistem ini sangat rentan terserang virus, penggunaan aplikasi untuk sistem ini rata-rata harus memiliki lisensi. Pada aplikasi dan bahasa pemrograman yang digunakan di-*install* satu per satu sesuai kebutuhan. Sehingga apabila terjadi kerusakan, maka dilakukan instalasi kembali di mulai dari awal. Hal ini membutuhkan waktu yang lama, serta membutuhkan perawatan yang baik terhadap virus dan kerusakan lainnya.

Adapun data lain yang didapatkan seperti bahasa pemrograman dan aplikasi yang digunakan didapat dari mata kuliah yang disediakan seperti :

1. Mata kuliah dasar-dasar pemrograman.

Mata kuliah ini mempelajari tentang dasar-dasar pemrograman, dimulai dari algoritma hingga bagaimana membentuk program sederhana. Walaupun materi banyak pada algoritma, namun implementasinya juga langsung melakukan praktek pemrograman. Implementasi aplikasi yang digunakan berdasarkan materi menggunakan bahasa pemrograman C dan Pascal.

2. Mata kuliah Algoritma dan Pemrograman

Mata kuliah ini merupakan kelanjutan terhadap mata kuliah daspro, praktikum pemrograman lebih banyak dilakukan. Implementasi aplikasi yang digunakan yaitu pemrograman pascal.

3. Mata kuliah Struktur data

Mata kuliah ini lebih kepada teknik dalam membuat program, lebih fokus kepada pembuatan program yang terstruktur. Implementasi aplikasi yang digunakan yaitu Pemrograman C++ dan Pascal.

4. Mata kuliah Pemrograman Visual

Mata kuliah ini memfokuskan pada teknik membuat program secara visual, yang memiliki antarmuka langsung dalam memprogram secara grafis. Implementasi aplikasi yang digunakan yaitu Visual basic 6.0.

5. Mata kuliah Pemrograman Berorientasi Objek

Mata kuliah ini mempelajari cara pemrograman secara *objek oriented*, Bahasa pemrograman yang digunakan adalah java, Implementasi aplikasi yang digunakan yaitu Java editor, netbeans.

#### 6. Mata kuliah Pemrograman Internet

Mata kuliah ini memfokuskan bagaimana memprogram untuk membuat suatu aplikasi berbentuk web. Bahasa pemrograman yang digunakan adalah *HTML, CSS, PHP 5.1*. Implementasi aplikasi yang digunakan adalah untuk editor menggunakan notepad, untuk perancangan *user interface* menggunakan *macromedia dreamweaver*.

#### 7. Mata kuliah Pemrograman Database

Terdiri dari basis data, sistem basis data, basis data terdistribusi. Inti dari mata kuliah ini adalah pengolahan data dengan membuat program data.

Implementasi aplikasi yang digunakan adalah mysql5.1 (bersifat konsol) dan SQL server 2000 (bersifat grafis).

### 4.2 Analisa Sistem Baru

Sistem baru yang akan dibangun menggunakan metode *remastering*, metode ini dapat memasukkan data-data baru kedalam sistem operasi yang akan dirancang ulang sesuai dengan kebutuhan. Sistem operasi yang menjadi target pada penelitian ini adalah ubuntu 9.10, didalam sistem operasi ini akan ditambahkan data-data berupa aplikasi dan bahasa pemrograman yang diperlukan untuk pembelajaran mata kuliah pemrograman, pengurangan data yang tidak diperlukan, selain itu pada sistem baru ini dilakukan penyesuaian *device* dengan sistem operasi.

Pemilihan sistem operasi linux ubuntu 9.10 dikarenakan sistem ini sudah stabil, baik dari segi *hardware*, maupun ketersediaan aplikasi yang sangat banyak untuk keperluan komputasi. Terutama *software-software* untuk keperluan pemrograman.

Tampilan pada sistem baru ini disesuaikan mirip seperti *windows xp* agar memudahkan dalam pemakaian sistem ini. Selain itu konfigurasi terhadap *hardware* tidak terlalu rumit dengan sistem deteksi yang baik.

Tahap analisa sistem baru ini juga meliputi analisa data-data aplikasi dan bahasa pemrograman yang diperlukan, metode yang digunakan untuk mengembangkan (pemaketan) aplikasi, *device* yang tersedia.

#### 4.2.1 Analisa data

Adapun data aplikasi dan bahasa pemrograman yang dibutuhkan untuk tiap-tiap mata kuliah melalui hasil wawancara adalah :

**Tabel 4.1 Daftar kebutuhan aplikasi pemrograman tiap mata kuliah**

Nama Mata kuliah	Bahasa Pemrograman yang digunakan sebelumnya dan aplikasi	Bahasa Pemrograman yang akan dipaketkan dan Aplikasi
Dasar-dasar pemrograman	<i>C dan Pascal</i>	<i>GNU C Compilers</i>
Algoritma dan Pemrograman	<i>Pascal</i>	<i>Free Pascal</i>
Struktur Data	<i>C++ dan Pascal</i>	<i>GNU C Compilers</i>
Pemrograman Visual	<i>Visual basic 6.0</i>	<i>Gambas 2.0</i>
Pemrograman Internet	<i>HTML, CSS, PHP 5.1.</i>	<i>CSS dan HTML editor(bluefish), PHP 5.1</i>
Pemrograman Berorientasi objek	<i>Java (java editor dan Netbeans)</i>	<i>Java (Eclipse)</i>
Pemrograman Basis data	<i>Mysq5.1 dan SQL server 2000</i>	<i>Mysql 5.1 dan Postgre 8.04</i>

Sedangkan untuk tambahan aplikasi adalah :

**Tabel 4.2 Daftar aplikasi tambahan dalam Pemrograman**

Bidang aplikasi	Aplikasi yang dipakai sebelumnya	Aplikasi yang akan dipaketkan
<i>Diagram Database</i>	<i>Kivio</i>	<i>Civio</i>
<i>Unified Modifying language (UML)</i>	<i>Rational Rose</i>	<i>Umbrello</i>
<i>Database</i>	<i>phpmyadmin</i>	<i>Phpmyadmin, pgAdmin III</i>

Sedangkan data aplikasi yang dibuang pada ubuntu 9.10 ini adalah :



**Tabel 4.3 Daftar Aplikasi yang dibuang**

Nama aplikasi	Keterangan
Games*	Permainan
Mplayer, sound recorder, brasero, mpalyer	<i>Sound and video</i>
gimp	Desain

#### **4.2.2 Analisa Metode *Remastering***

Metode *Remastering* yaitu teknik perancangan distribusi GNU Linux dengan cara memecah isi (*unpack iso*) sistem operasi GNU Linux kedalam bentuk struktur direktori, tiap-tiap struktur direktori dikonfigurasi berdasarkan fungsinya, baik berupa *remove, add, update, upgrade, plan modul and driver*, dan hal lainnya, yang langsung dijalankan ketika proses terjadi, kemudian dibentuk kembali menjadi *file iso*.

Metode *remastering* cocok bagi pemula yang ingin mengembangkan linux dengan syarat :

1. Distro yang dipilih memiliki basis dukungan yang kuat, baik dari korporat ataupun dari komunitas.
2. Memiliki ketersediaan *repository* (aplikasi) yang lengkap.
3. Mudah melakukan *update*.

#### **4.2.3 Analisa *Hardware***

Analisa *hardware* diperlukan untuk melakukan *compile* kernel, dengan mendapatkan data-data *hardware* secara lengkap, maka akan dengan mudah melakukan *compile* kernel langsung. Tujuan dari *compile* kernel adalah agar sistem dapat mengenali *hardware* yang ada dan *load driver* secara cepat, sehingga sistem tidak terlalu lama memproses *module* yang akan digunakan.

**Tabel 4.4 Informasi *Hardware***

Nama <i>hardware</i>	keterangan
Processor	Intel ® Pentium ® dual cpu e 2180 @ 2.00 GHz
Ram	DIMM SDRAM 512 MB
Dvd-rom	Dvd-RAM writer
Ethernet	Realtek Semiconductor Co
Harddisk	ATA Disk
Motherboard	Wearnes premier 8310l
Pci bride 1	Intel Corporation 82801g (ICH7) Family
Pci bride 2	Intel Corporation 82801g (ICH7) Family
Network controller	RAlink

Pada linux ubuntu untuk mengetahui *device* ketik `lspci` di terminal :

```
ubuntu@ubuntu:~$ lspci

00:00.0 Host bridge: Intel Corporation 82G33/G31/P35/P31 Express DRAM Controller (rev 10)
00:02.0 VGA compatible controller: Intel Corporation 82G33/G31 Express Integrated Graphics Controller (rev 10)
00:1b.0 Audio device: Intel Corporation 82801G (ICH7 Family) High Definition Audio Controller (rev 01)
00:1c.0 PCI bridge: Intel Corporation 82801G (ICH7 Family) PCI Express Port 1 (rev 01)
00:1c.1 PCI bridge: Intel Corporation 82801G (ICH7 Family) PCI Express Port 2 (rev 01)
00:1d.0 USB Controller: Intel Corporation 82801G (ICH7 Family) USB UHCI Controller #1 (rev 01)
00:1d.1 USB Controller: Intel Corporation 82801G (ICH7 Family) USB UHCI Controller #2 (rev 01)
00:1d.2 USB Controller: Intel Corporation 82801G (ICH7 Family) USB UHCI Controller #3 (rev 01)
00:1d.3 USB Controller: Intel Corporation 82801G (ICH7 Family) USB UHCI Controller #4 (rev 01)
00:1d.7 USB Controller: Intel Corporation 82801G (ICH7 Family) USB2 EHCI Controller (rev 01)
00:1e.0 PCI bridge: Intel Corporation 82801 PCI Bridge (rev e1)
00:1f.2 IDE interface: Intel Corporation 82801GB/GR/GH (ICH7 Family) SATA IDE Controller (rev 01)
```

**Gambar 4.1 *device* computer**

### 4.3 Perancangan Remastering Linux Ubuntu 9.10

Tahap selanjutnya adalah tahapan perancangan, dilakukan setelah data-data terkumpul dan analisa dilakukan, tahapan perancangan pada *remastering* distro linux ubuntu 9.10 ini adalah dengan memberikan gambaran berupa alur proses yang dilakukan ketika *remastering*.

Alur proses pada perancangan haruslah selesai setiap tahap tanpa adanya, *error* yang terjadi pada tahap tersebut, karena apabila terdapat *error* pada tahap perancangan akan menyebabkan sistem tidak akan berjalan dengan baik.

Tahapan-tahapan yang dilalui dalam melakukan *remastering* ubuntu 9.10 adalah :

1. Tahap Persiapan

Pada tahapan ini, disediakan beberapa *tools* pembantu dalam proses remastering seperti paket *squashfs-tools*, paket *genisoimage*, *gnu c compiler* dan iso ubuntu 9.10

2. Tahap *Decompress*

Pada tahapan ini, seluruh iso ubuntu akan di-*extract* ke sebuah direktori dimana salah satu *file* yang di-*extarct* adalah *filesystem* default ubuntu yaitu *file system.squashfs* yang masih berbentuk *file compress*.

3. Tahap *Decompress filesystem*

Tahap ini merupakan tahap *extract* seluruh isi *file filesystem.squashfs* yang berisi *filesystem default* ubuntu.

4. Tahap Persiapan *Mode chroot*

Tahap mode *chroot* merupakan tahap mempersiapkan *file-file* yang akan dibutuhkan seperti bentuk *file biner*, *file deb*, *file tar.gz*.

5. Tahap *chroot*

Tahap memasuki *filesystem* baru, tahap ini berada di luar lingkungan linux ubuntu komputer induk.

6. Tahap Kustomisasi

Merupakan tahap perubahan pada iso linux hasil *extract*, seperti membuang aplikasi, menambah, *update* dan perubahan pada *theme*, *background*, *panel*, dan juga *icon*.

7. Tahap *Cleaning Up*

Merupakan tahap pembersihan dan *cache* maupun *temporary* hasil *install* dan *remove software* saat tahapan kustomisasi.

8. Tahap *Manifest*

Pembentukan *file manifest* untuk menjadi iso.

9. Tahap *Compress file Squashfs*

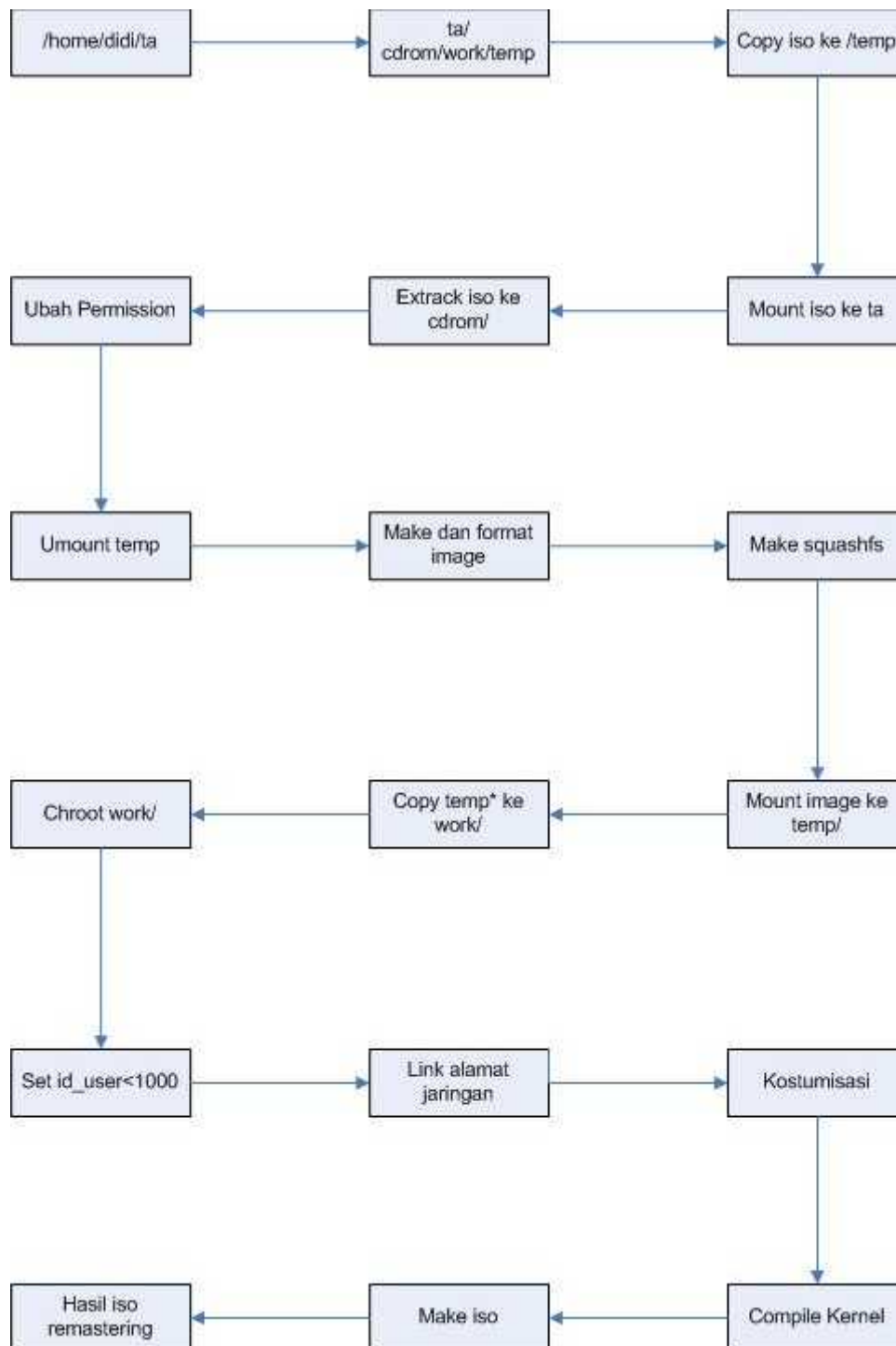
Pembentukan *file filesystem.squashfs*

10. Tahap *Compress ISO*

Tahap akhir pembentukan ISO.

Berikut ini merupakan rancangan proses *remastering* yang penulis lakukan dari tahap awal sampai tahap akhir :

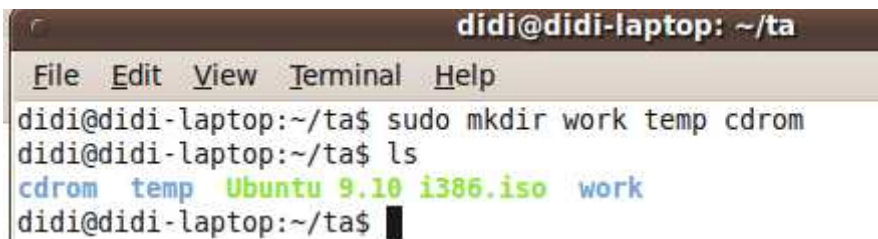
Alur proses rancangan *remastering* ubuntu 9.10 secara manual :



Gambar 4.2 Alur Proses Rancangan *Remastering* Distro Linux 9.10

Proses perancangan secara detail pada *remastering* distro linux ubuntu 9.10 yang penulis lakukan secara bertahap sebagai berikut : pertama buat 1 direktori bernama ta dan 3 direktori tempat *remastering* linux ubuntu 9.10 dan *copy* iso ubuntu 9.10 ke folder ta.

```
# mkdir cdrom temp work
```

A screenshot of a terminal window titled 'didi@didi-laptop: ~/ta'. The terminal shows the following commands and output: 'didi@didi-laptop:~/ta\$ sudo mkdir work temp cdrom', 'didi@didi-laptop:~/ta\$ ls', and the output 'cdrom temp Ubuntu 9.10 i386.iso work'. The prompt 'didi@didi-laptop:~/ta\$' is followed by a cursor.

```
didi@didi-laptop: ~/ta
File Edit View Terminal Help
didi@didi-laptop:~/ta$ sudo mkdir work temp cdrom
didi@didi-laptop:~/ta$ ls
cdrom temp Ubuntu 9.10 i386.iso work
didi@didi-laptop:~/ta$
```

Gambar 4.3 *copy iso*

```
# sudo mount -o loop ubuntu 9.10 /temp
```

Lakukan *mount* iso ke *folder* sementara yaitu *temp folder*.

A screenshot of a terminal window titled 'didi@didi-laptop: ~/ta'. The terminal shows the following commands and output: 'didi@didi-laptop:~/ta\$ ls', 'cdrom temp Ubuntu 9.10 i386.iso work', and 'didi@didi-laptop:~/ta\$ sudo mount -o loop ubuntu\ 9.10\ i386.iso temp/'. The prompt 'didi@didi-laptop:~/ta\$' is followed by a cursor.

```
didi@didi-laptop: ~/ta
File Edit View Terminal Help
didi@didi-laptop:~/ta$ ls
cdrom temp Ubuntu 9.10 i386.iso work
didi@didi-laptop:~/ta$ sudo mount -o loop ubuntu\ 9.10\ i386.iso temp/
didi@didi-laptop:~/ta$
```

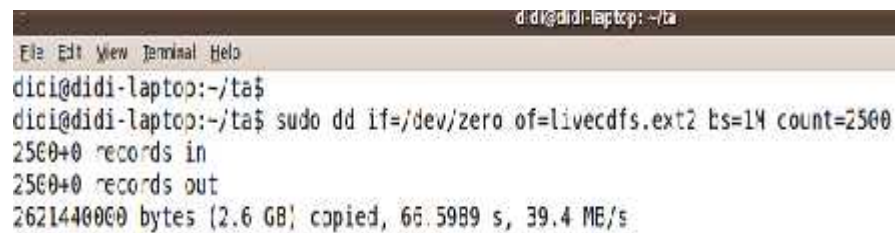
Gambar 4.4 *mount iso*

Kemudian ekstrak iso kedalam *cdrom folder*, ubah hak izin di *folder cdrom* dan lepaskan kembali kaitan pada *folder temp*. Pada ekstrak iso di *temp* ke *cdrom* isinya dapat dilihat di lampiran A.

```
# sudo rsync -av temp/ cdrom
```

Setelah itu buat *virtual image* dengan melakukan *disk dump* dengan ukuran pada blok awal 1Mb dan blok akhir 2500 kemudian *format image* tersebut dengan memanggil fungsi *format image* yaitu *mke2fs*.

```
# sudo dd if=/dev/zero of=livedfs.ext2 bs=1M
```

A screenshot of a terminal window with a dark background. The title bar at the top reads 'didi@didi-laptop: ~/ta'. Below the title bar is a menu bar with 'File Edit View Terminal Help'. The terminal shows the following text: 'didi@didi-laptop:~/ta\$ sudo dd if=/dev/zero of=livedfs.ext2 bs=14 count=2500', followed by '2500+0 records in', '2500+0 records out', and '262144000 bytes (2.6 GB) copied, 66.5989 s, 39.4 MB/s'.

```
didi@didi-laptop:~/ta$ sudo dd if=/dev/zero of=livedfs.ext2 bs=14 count=2500
2500+0 records in
2500+0 records out
262144000 bytes (2.6 GB) copied, 66.5989 s, 39.4 MB/s
```

Gambar 4.5 *make image*

Memanggil *format image mke2fs* sebagai format standar linux. Kemudian *virtual image* siap diproses dengan memberikan data (seperti *type* sistem operasi tempat diformat, *Block size*, *Fragments size*, *jumlah inodes*, *maximum blok*) data tersebut sangat penting untuk mengetahui seberapa besar *image* yang akan digunakan dalam proses *remastering* seperti jika ingin menambah seberapa besar aplikasi yang akan dikompilasi pada sebuah distro dengan mengukur daya tampung *image* yang dibuat.

```
# sudo mke2fs livedfs.ext2-
```

```
didi@didi-laptop: ~/ta
File Edit View Terminal Help
didi@didi-laptop:~/ta$
didi@didi-laptop:~/ta$ sudo mke2fs livecdfs.ext2
mke2fs 1.41.9 (22-Aug-2009)
livecdfs.ext2 is not a block special device.
Proceed anyway? (y,n) y
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
150600 inodes, 640000 blocks
32000 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=655365728
20 block groups
32768 blocks per group, 32768 fragments per group
8000 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 25 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

Gambar 4.6 *Format Image*

Tahap selanjutnya buat *folder squashfs* sebagai tempat ekstrak desktop sistem dan *mount* ke cdrom. Untuk *image livecdfs.ext2* *mount* ke *folder work*, dan *umount temp folder*.

```
# sudo mkdir squashfs
```

```
# sudo mount -t squashfs -o loop
cdrom/casper/filesystem.squashfs temp/
```

```
didi@didi-laptop: ~/ta
File Edit View Terminal Help
didi@didi-laptop:~/ta$
didi@didi-laptop:~/ta$ sudo mkdir squashfs
didi@didi-laptop:~/ta$ sudo mount -t squashfs -o loop cdrom/casper/filesystem.squashfs temp/
```

Gambar 4.7 *make squashfs*

Selanjutnya *copy* isi *temp* ke *work*, karena jumlahnya banyak hasil *copy* yang telah diesktrak dapat dilihat dilampiran B.



Kemudian masuk ke *chroot*, posisi terminal yang akan digunakan untuk *remastering*, bukan sebagai terminal utama. Di terminal *chroot* ini, dapat dilihat iso hasil ekstrak kedalam direktori linux.

Setelah itu edit *user\_id*, sewaktu proses *remastering*, tidak boleh ada *user id* bernilai >1000. Oleh sebab itu, *Set id user* dengan *syntak* berikut :

```
# grep '^[^:]*:[^:]*:[1-9][0-9][0-9][0-9]:' /etc/passwd
```

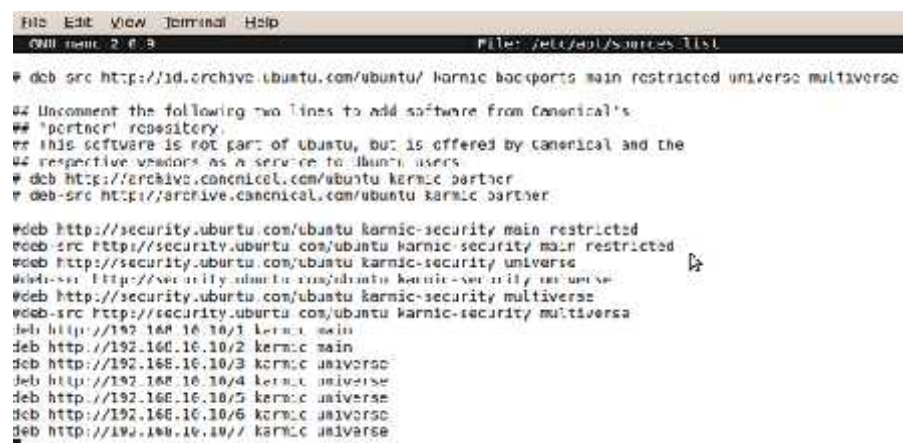
```
# grep '^[^:]*:[^:]*:[12][0-9][0-9][0-9]:' /etc/passwd
```

Jika mendapatkan banyak UID >=1000, rubah UID-nya

```
# usermod -u 500 $hit
```

Setting *user id* dapat dilihat di lampiran C, Tahap selanjutnya link terminal *chroot* ke *host* jaringan dan lakukan *update*, edit file nano */etc/apt/sources.list* sesuaikan alamat file repo dan jenisnya.

Konfigurasi link jaringan ke alamat ubuntu *repository* untuk *update* data-data yang diperlukan. Pada perancangan *remastering* ubuntu 9.10 ini penulis membuat jaringan *client-server* lokal.

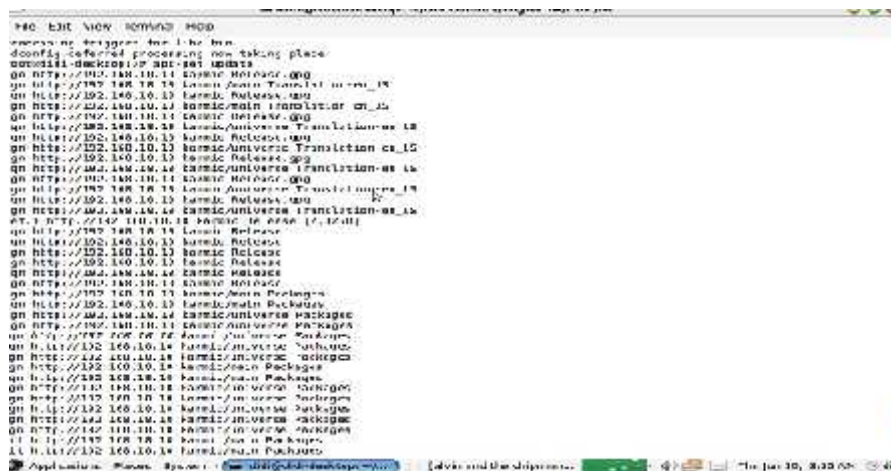


```
File Edit View Terminal Help
GNU nano 2.6.3 File: /etc/apt/sources.list

# deb-src http://id.archive.ubuntu.com/ubuntu/ karmic backports main restricted universe multiverse
## Uncomment the following two lines to add software from Canonical's
## 'partner' repository.
## This software is not part of Ubuntu, but is offered by Canonical and the
## respective vendors as a service to Ubuntu users.
# deb http://archive.canonical.com/ubuntu karmic partner
# deb-src http://archive.canonical.com/ubuntu karmic partner

#deb http://security.ubuntu.com/ubuntu karmic-security main restricted
#deb-src http://security.ubuntu.com/ubuntu karmic-security main restricted
#deb http://security.ubuntu.com/ubuntu karmic-security universe
#deb-src http://security.ubuntu.com/ubuntu karmic-security universe
#deb http://security.ubuntu.com/ubuntu karmic-security multiverse
#deb-src http://security.ubuntu.com/ubuntu karmic-security multiverse
deb http://192.168.16.10/1 karmic main
deb http://192.168.16.10/2 karmic main
deb http://192.168.16.10/3 karmic universe
deb http://192.168.16.10/4 karmic universe
deb http://192.168.16.10/5 karmic universe
deb http://192.168.16.10/6 karmic universe
deb http://192.168.16.10/7 karmic universe
```

Gambar 4.8 Link ke server lokal



Gambar 4.9. *Update* alamat repo ke terminal utama

Tahap ini merupakan tahap *remastering* dengan menambah aplikasi, membuang, mengubah *theme*, *icon*, *wallpaper background* dan lainnya. Aplikasi yang di *update* pada *remastering* ini disesuaikan pada hasil analisis aplikasi pemograman.

```
# apt-get install <nama aplikasi>
```

```
# apt-get remove <nama aplikasi>
```

Sebelum proses pembentukan iso, penulis melakukan *compile kernel*, proses *compile kernel* dapat dilihat pada lampiran D.

Setelah selesai di konfigurasi aplikasi yang dibutuhkan selanjutnya dibentuk kembali menjadi iso. Berikut proses yang dilalui :

```
$ sudo chmod +w cdrom/casper/filesystem.manifest
```

Selanjutnya *update* aplikasi yang telah di *install* dengan *syntak* berikut :

```
$ sudo chroot work dpkg-query -W -
showformat='${Package}${version}\n'>cdrom/casper/filesyste
m.manifest
```

Copy *filesystem.manifest* menjadi *filesystem.manifest-desktop*

```
$ sudo cp cdrom/casper/filesystem.manifest
cdrom/casper/filesystem-manifest
```

Selanjutnya membuat *installer* dengan menggunakan *ubiquity*, *installer* ini akan disediakan ketika awal *booting* system.

```
$ sudo sed -i '/ubiquity/d'
cdrom/casper/filesystem.manifest-desktop
```

Hapus *file squashfs* lama

```
$ sudo rm cdrom/casper/filesystem.squashfs
```

Selanjutnya buat *squashfs* dengan *mksquashfs*

```
$ sudo mksquashfs work cdrom/casper/filesystem.squashfs
```



```
didi@didid-desktop: ~/ta2
$ sudo mksquashfs work cdrom/casper/filesystem.squashfs
Creating 4.0 filesystem on cdrom/casper/filesystem.squashfs, block size 131072.
=====
Exportable Squashfs 4.0 filesystem, data block size 131072
compressed data, compressed metadata, compressed fragments
duplicates are removed
Filesystem size 923874.17 Kbytes (933.47 Mbytes)
41.76% of uncompressed filesystem size (2292267.01 Kbytes)
Inode table size 1421814 bytes (1387.71 Kbytes)
25.32% of uncompressed inode table size (5611354 bytes)
Directory table size 1219225 bytes (1207.88 Kbytes)
19.49% of uncompressed directory table size (6166661 bytes)
Number of duplicate files found 11131
Number of inodes 146525
Number of files 184316
Number of fragments 6768
Number of symbolic links 26186
Number of device nodes 37
Number of fifo nodes 2
Number of socket nodes 0
Number of directories 13963
Number of ids (unique uids + gids) 33
Number of uids 18
  root [0]
  hplip [183]
  san [6]
  daemon [1]
  cronddh [186]
  avahi-daemon [284]
  speech-dispatcher [188]
```

Gambar 4.10 Proses membuat *squashfs* dengan *mksquashfs*

Selanjutnya hapus *md5sum* lama

```
$ sudo rm cdrom/md5sum.txt
```

Dan *copy* md5sum yang ada di *cdrom* folder

```
$Sudo find -type f -print0|xargs -0 md5sum| sudo tee  
md5sum.txt
```

Proses hasil *copy md5sum.txt* yang baru dapat dilihat di lapiran E

Kemudian tahap akhir yaitu tahap pembentukan iso :

```
$ sudo mkisofs -D -r -V ubuntu9_10.iso -cache-inodes -J -l  
-b isolinux/isolinux.bin -c isolinux/boot.cat -no-emul-  
boot-load-size 4- boot-info-table-o ubuntu9_10.iso  
/home/didi/ta6/cdrom
```

Proses pembentukan iso ini dapat dilihat di lampiran F

## BAB V

### IMPLEMENTASI DAN PENGUJIAN

#### 5.1 Implementasi

Implementasi merupakan tahap perangkat lunak siap dioperasikan pada keadaan yang sebenarnya, sehingga akan diketahui apakah perangkat lunak yang dibuat benar-benar dapat menghasilkan tujuan yang ingin dicapai.

#### 5.2 Batasan Implementasi

Batasan implementasi pada penulisan Tugas Akhir ini adalah :

1. Menampilkan hasil uji dari instalasi sistem operasi Ubuntu *Remastering*.
2. Menampilkan aplikasi pemrograman secara liveDVD.

#### 5.3 Lingkungan Implementasi

##### Perangkat Keras

1. *Processor* : Intel dual core, 2.00 GHz
2. *Memory* : 512 MB
3. *Harddisk* : 160 GB

##### Perangkat Lunak

1. Sistem Operasi : *Ubuntu 9.10 Programming*
2. Aplikasi Bahasa Pemrograman *Open Source*
3. Kernel : 2.6.31-generic

#### 5.4 Hasil Implementasi

Berikut ini merupakan hasil aplikasi dari *remastering* ubuntu 9.10 dengan memperbandingkan sistem operasi asalnya yaitu ubuntu 9.10.

**Tabel 5.1 Perbandingan Aplikasi antara ubuntu 9.10 dengan ubuntu *programming*.**

Nama aplikasi/bahasa pemrograman	Ubuntu 9.10	Ubuntu 9.10 <i>Programming</i>
Openoffice.org word processor	Ada	ada
Openoffice.org spreadsheet processor	Ada	ada
Gnu C compiler	Ada	ada
Free Pascal	Tidak ada	Ada
Bluefish	Tidak ada	Ada
Mysql 5.1	Tidak ada	ada
Apache 2	Tidak ada	ada
Gambas 2	Tidak ada	ada
Eclipse	Tidak ada	ada
Php5.1	Tidak ada	ada
PostgreSQL 8.04	Tidak ada	ada
Firefox	Ada	ada
Gimp	Ada	Tidak ada
Games*	Ada	Tidak ada
Brasero	Ada	Tidak ada
Movie player	Ada	Tidak ada
Rhythmbox	Ada	Tidak ada
Sound Recorder	Ada	Tidak ada

Implementasi ini akan menampilkan desktop sistem operasi dan aplikasi pemograman yang telah di *install* sewaktu proses *remastering* antara lain :

## 1. Implementasi *antar muka desktop*

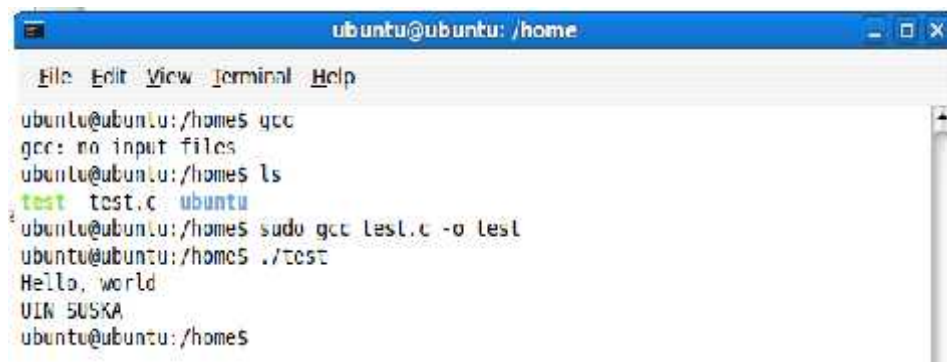
Berikut ini adalah gambar *desktop ubuntu programmer*, desktop sudah dimodifikasi seperti *theme*, *panel*, *icon*. Pemilihan *theme* ini agar memudahkan dalam pencarian menu aplikasi.



Gambar 5.1 Desktop Ubuntu Programmer

## 2. Impementasi *Aplikasi gcc Compiler*

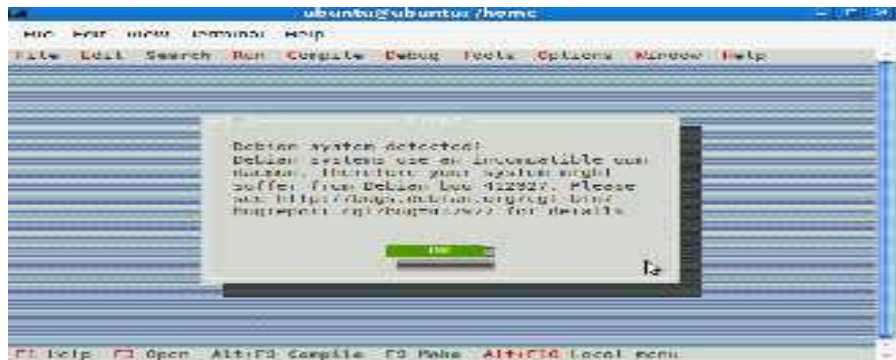
Berikut ini adalah gambar *antar muka GNU C compiler*, pemanggilan melalui konsol, dengan mengetik gcc dikonsol.



Gambar 5.2 Antar muka gcc c compiler

### 3. Impementasi Aplikasi *Free Pascal*

Pemanggilan aplikasi ini melalui konsol dengan mengetik fp, maka akan keluar antar muka sebagai berikut :



**Gambar 5.3** Antar muka Free Pascal compiler

### 4. Impementasi Aplikasi *Gambas2*

Pemanggilan aplikasi ini melalui konsol dengan mengetik gambas atau pada star menu pada menu programming, maka akan keluar antar muka sebagai berikut :

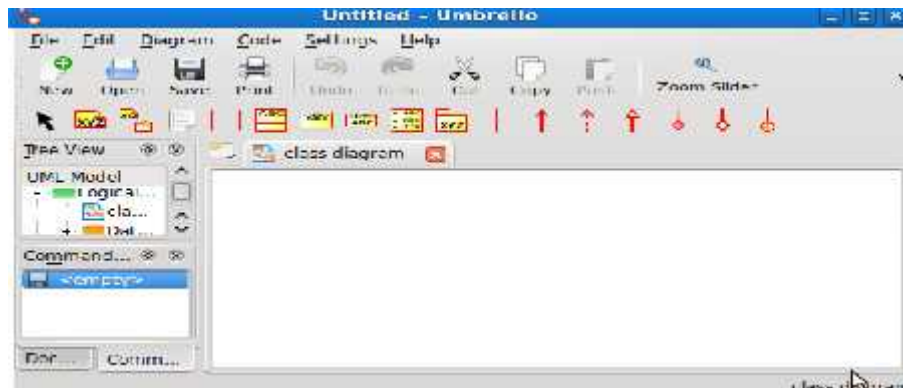


**Gambar 5.4** Antar muka gambas2



## 5. Impementasi Aplikasi *Umbrello*

Pemanggilan aplikasi ini melalui konsol atau pada start menu pada menu programming, maka akan keluar antar muka sebagai berikut :



Gambar 5.5 Antar muka Umbrello

## 6. Impementasi bahasa pemrograman *php*

Untuk menguji bahwa pada sistem ini sudah ada aplikasi yang mendukung pemrograman php5.1, dibuat contoh coding seperti :

```
<? Infophp();  
?>
```

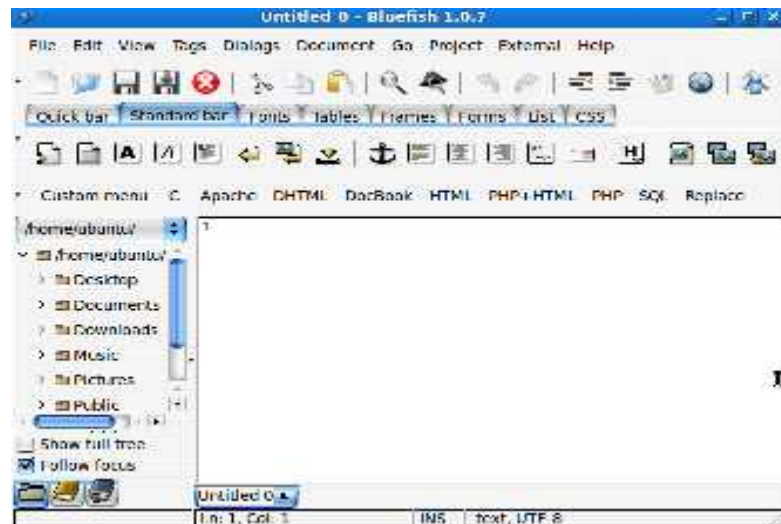
Simpan dengan nama info.php di folder /var/www/ dan buka web browser, ketikan localhost/info.php maka akan muncul sebagai berikut :

<b>System</b>	Linux ubuntu 2.6.31-14-generic #48 Ubuntu SMP Fri Oct 11 14:04:25 UTC 2009; i686
<b>Build Date</b>	Oct 23 2009 16:04:10
<b>Server API</b>	Apache 2.0 Handler
<b>Virtual Directory Support</b>	disabled
<b>Configuration File (php.ini) Path</b>	/etc/php5/apache2
<b>Loaded Configuration File</b>	/etc/php5/apache2/php.ini
<b>Scan this dir for additional .ini files</b>	/etc/php5/apache2/conf.d
<b>additional .ini files parsed</b>	/etc/php5/apache2/conf.d/gd.ini, /etc/php5/apache2/conf.d/mcrypt.ini, /etc/php5/apache2/conf.d/mysql.ini, /etc/php5/apache2/conf.d/pgsql.ini,

**Gambar 5.6 Antar muka info.php**

**7. Impementasi Aplikasi *bluefish***

Pemanggilan aplikasi ini melalui konsol dengan mengetik bluefish atau pada start menu, maka akan keluar antar muka sebagai berikut :



**Gambar 5.7 Gambar bluefish**

**8. Impementasi Aplikasi pgAdmin**

Pemanggilan aplikasi ini melalui konsol dengan mengetik pgAdmin atau pada start menu, maka akan keluar antar muka sebagai berikut :



## 5.7 Hasil Pengujian

Pengujian sistem merupakan pengujian terhadap instalasi sistem operasi dan pengujian aplikasi pemrograman.

**Tabel 5.2 Pengujian Instalasi Sistem Menggunakan Black Box**

Jenis uji	Keadaan	Keterangan
Mode menu instalasi	Tampilan menu instalasi	ada
Keyboard	Tes ketikan	sukses
Harddisk	Partisi harddisk	sukses
Mouse	Pergerakan Mouse	sukses
Mode Desktop	Tampilan Desktop	ada
Tampilan menu	Menu aplikasi	ada

**Tabel 5.3 Pengujian Aplikasi Pemrograman Menggunakan Black Box**

Nama aplikasi/bahasa pemrograman	Compile coding	Hasil yang diharapkan	Hasil Tes
Gcc compiler	berjalan	Sukses	Sukses
Gambas2	berjalan	Sukses	Sukses
Free pascal	Berjalan	Sukses	Sukses
Bluefish	berjalan	Sukses	Sukses
Eclipse	berjalan	Sukses	Sukses
Umbrella	berjalan	Sukses	Sukses
Apache2	berjalan	Sukses	Sukses
Php5.1	berjalan	Sukses	Sukses
PgAmin	berjalan	Sukses	Sukses

## 5.8 Kesimpulan Pengujian

Pada pengujian sistem operasi ubuntu 9.10 *Programming* ini penulis menemukan adanya kesalahan pada proses *remastering* ubuntu 9.10 seperti :

1. Tidak bisa tampil xsplash pada pemilihan menu *boot*, sehingga warna boot menu tampak hitam pada *background*, hal ini disebabkan perubahan ubuntu 9.10 pada grub yang menggunakan isolinux. Sehingga xsplash pada ubuntu remastering ini di *remove*, untuk mengurangi beban memori sewaktu *boot*.
2. Seluruh aplikasi tambahan berupa aplikasi *programming*, dapat berjalan secara *livedvd*.

## BAB VI

### PENUTUP

#### 6.1 Kesimpulan

Berdasarkan hasil studi literatur dan eksperimen yang dilakukan pada penelitian ini, maka dapat disimpulkan hal-hal sebagai berikut:

1. Sistem operasi ubuntu 9.10 ini telah berhasil ditanamkan aplikasi pemrograman untuk media pembantu kegiatan pembelajaran pemrograman *open source* di jurusan teknik informatika UIN Suska Riau.
2. Penerapan teknik *remastering* pada sistem operasi ubuntu 9.10, memudahkan dalam menambahkan dan mengatur kebutuhan aplikasi yang diinginkan.
3. Sistem operasi ini belum bisa mendeteksi *hardware* selain yang terdapat pada penelitian ini disebabkan adanya kompilasi pada kernel.
4. Ukuran sistem menjadi besar disebabkan aplikasi yang ditanam cukup memakan memori, sehingga sistem akan menjadi lambat ketika *booting*.

## 6.2 Saran

Sebagai tindak lanjut dari penelitian, dikemukakan beberapa saran sebagai berikut:

1. Agar sistem mampu mendeteksi jenis *hardware* terbaru, perlu ditanamkan *update driver* terbaru dari *hardware* tersebut.
2. Perlu adanya penelitian lebih lanjut mengenai aplikasi pemograman yang dapat ditambahkan pada sistem operasi dan dijadikan sebagai media pembelajaran.
3. Ukuran sistem operasi menjadi lebih besar, sehingga beban memori bertambah, perlu di cari aplikasi pembanding yang lebih kecil ukurannya tetapi bisa digunakan sebagai media pembelajaran pemograman.

## DAFTAR PUSTAKA

- Andreas, J ; *Pengenalan Sistem Operasi Linux*. INDAH Surabaya, Surabaya, 2003.
- Hariningsih, SP ; *Sistem Operasi*. Graha Ilmu, Yogyakarta, 2003.
- Hariyanto, Bambang; *Sistem Operasi*. Informatika Bandung, Bandung, 2006.
- Pressman, Roger S; *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku Satu)* , ANDI Yogyakarta, Yogyakarta, 2002.
- Sofyan, Ahmad; *Membuat Distro Linux Sendiri*. Jakarta: Dian Rakyat. 2006.
- Stallings William; *Operating Systems: Internal and Design Principles*. Fourth Edition, Prentice-Hall International, New Jersey, 2001.
- Tanenbaum, Andrew S; *Modern Operating System Second Edition*. New Jersey; Prentice-Hall, Inc 2001.
- \_\_\_\_\_, Kelebihan dan Kekurangan Linux [Online] Available ; <http://www.indowebster.web.id/archive/index.php?t-29296.html> diakses tanggal 11 Agustus 2010.
- \_\_\_\_\_, *Kernel Linux [online] available*; <http://www.kernel.org> diakses tanggal 01 juni 2010.
- \_\_\_\_\_, Struktur Sistem Linux [Online] Available; <http://www.scribd.com/doc/19529837/Struktur-Sistem-Operasi-Pada-Linux> diakses tanggal 11 Agustus 2010.
- \_\_\_\_\_, *Open Source Software (OSS)[online] available*; [http://www.dwheeler.com/off\\_fs\\_why.html](http://www.dwheeler.com/off_fs_why.html), diakses tanggal 10 September 2009.
- \_\_\_\_\_, *Remastering Ubuntu Manual, [online] available*; <http://www.wiki-wiki.org> diakses tanggal 10 maret 2010.